Amendments to the Claims:

This listing of claims will replace all prior versions, and listings, of claims in the application:

1. (Currently amended) A hybrid ~~serial/parallel~~ serial to parallel bus interface comprising: a data block demultiplexing device having an input configured to receive a data block and demultiplexing the data block into a plurality of nibbles, each nibble having a plurality of bits; for each nibble: a parallel to serial converter for converting that nibble into serial data; a line for transferring that nibble serial data; and a serial to parallel converter for converting that nibble serial data to recover that nibble; and a data block reconstruction device for combining the recovered nibbles into the data block.

2. (Original) The interface of claim 1 wherein a number of bits in a data block is N and a number of the lines is i and $1<i<N$.

3. (Original) The interface of claim 1 wherein a number of bits in a nibble is four and a number of lines is two.

4. (Currently amended) A hybrid ~~serial/parallel~~ serial to parallel bus interface comprising: means having an input configured to receive a data block for

demultiplexing the data block into a plurality of nibbles, each nibble having a plurality of bits; for each nibble: means for converting that nibble into serial data; a line for transferring that nibble serial data; and means for converting that nibble serial data to recover that nibble; and means for combining the recovered nibbles into the data block.

5. (Original) The interface of claim 4 wherein a number of bits in a data block is N and a number of the lines is i and 1<i<N.

6. (Original) The interface of claim 4 wherein a number of bits in a nibble is four and a number of lines is two.

7. (Original) A method for transferring data, the method comprising: providing a data block; demultiplexing the data block into a plurality of nibbles, each nibble having a plurality of bits; for each nibble: converting that nibble into serial data; providing a line and transferring the nibble serial data over the line; converting that nibble serial data into parallel data to recover that nibble; and combining the recovered nibbles into the data block.

8. (Original) The method of claim 7 wherein a number of bits in a data block is N and a number of the lines is i and 1<i<N.

9. (Original) The method of claim 7 wherein a number of bits in a nibble is four and a number of lines is two.

10. (Original) A method for transferring a data block through an interface connecting a first node to a second node, the method comprising: demultiplexing the data block into m sets of n bits; adding a start bit to each of the m sets, the m start bits collectively representing one of a particular mathematical function or destination; transferring from the first node each of the m sets over a separate line; receiving at the second node each of the transferred m sets; and utilizing the received m sets in accordance with the m start bits.

11. (Original) The method of claim 10 <u>further</u> comprising:

wherein at least one of the m start bits being in a one state and when the interface is not transmitting data, all the separate lines being in a zero state.

12. (Original) The method of claim 10 wherein the m start bits represent a start of data transfer.

13. (Original) The method of claim 10 wherein the m start bits collectively represent a particular mathematical function and not a destination.

14. (Original) The method of claim 10 wherein functions that the m start bits collectively represent include a relative increase, a relative decrease and an absolute value functions.

15. (Original) The method of claim 10 wherein the m start bits collectively represent a particular destination and not a mathematical function.

16. (Original) The method of claim 15 wherein destinations that the m start bits collectively represent include a RX receive and TX transmit gain controller.

17. (Currently amended) ~~The method of claim 10 wherein the m start bits collectively represent both a particularly mathematical function and a particular destination~~. A method for transferring a data block through an interface connecting a first node to a second node, the method comprising: demultiplexing the data block into m sets of n bits; adding a start bit to each of the m sets, the m start

bits collectively representing both a particular mathematical function and destination; transferring from the first node each of the m sets over a separate line; receiving at the second node each of the transferred m sets; and utilizing the received m sets in accordance with the m start bits.

18. (Currently amended) A hybrid ~~serial/parallel~~ serial to parallel bus interface for transferring data from a first node to a second node, the interface comprising: a data block demultiplexing device for demultiplexing a data block from the first node into m sets of n bits and for adding a start bit to each of the m sets, the m start bits collectively representing one of a particular mathematical function or destination; for each of the m sets, a separate line for transferring that set of the m sets from the first node to the second node; a data block reconstruction device for receiving the m sets, for combining the m sets into the data block and for utilizing the m sets in accordance with the m start bits.

19. (Original) The interface of claim 18 wherein at least one of the m start bits being in a one state and when the interface is not transmitting data, all the separate lines being in a zero state.

20. (Original) The interface of claim 18 wherein the m start bits represent a start of data transfer.

21. (Original) The interface of claim 18 wherein the m start bits collectively represent a particular mathematical function and not a destination.

22. (Original) The interface of claim 18 wherein functions that the m start bits collectively represent include a relative increase, a relative decrease and an absolute value functions.

23. (Original) The interface of claim 18 wherein the m start bits collectively represent a particular destination and not a mathematical function.

24. (Currently amended) The interface of claim 23 wherein destinations that the m start bits collectively represent include a [[RX]] receive and [[TX]] transmit gain controller.

25. (Currently amended) ~~The interface of claim 18 wherein the m start bits collectively represent both a particular mathematical function and a particular destination~~. A hybrid serial to parallel bus interface for transferring data from a

first node to a second node, the interface comprising: a data block demultiplexing device for demultiplexing a data block from the first node into m sets of n bits and for adding a start bit to each of the m sets, the m start bits collectively representing both a particular mathematical function and destination; for each of the m sets, a separate line for transferring that set of the m sets from the first node to the second node; and a data block reconstruction device for receiving the m sets, for combining the m sets into the data block and for utilizing the m sets in accordance with the m start bits.

26. (Original) A hybrid ~~serial/parallel~~ serial to parallel bus interface for transferring data from a first node to a second node, the interface comprising: means for demultiplexing a data block into m sets of n bits; means for adding a start bit to each of the m sets, the m start bits collectively representing one of a particular mathematical function or destination; means for transferring from the first node each of the m sets over a separate line; means for receiving at the second node each of the transferred m sets; and means for utilizing the received m sets in accordance with the m start bits.

27. (Original) The interface of claim 26 wherein at least one of the m start bits being in a one state and when the interface is not transmitting data, all the separate lines being in a zero state.

28. (Original) The interface of claim 26 wherein the m start bits represent a start of data transfer.

29. (Original) The interface of claim 26 wherein the m start bits collectively represent a particular mathematical function and not a destination.

30. (Original) The interface of claim 26 wherein functions that the m start bits collectively represent include a relative increase, a relative decrease and an absolute value functions.

31. (Original) The interface of claim 26 wherein the m start bits collectively represent a particular destination and not a mathematical function.

32. (Currently amended) The interface of claim 31 wherein destinations that the m start bits collectively represent include a [[RX]] receive and [[TX]] transmit gain controller.

33. (Currrently amended) ~~The interface of claim 26 wherein the m start bits collectively represent both a particular mathematical function and a particular destination~~. A hybrid serial to parallel bus interface for transferring data from a first node to a second node, the interface comprising: means for demultiplexing a data block into m sets of n bits; means for adding a start bit to each of the m sets, the m start bits collectively representing both a particular mathematical function and destination; means for transferring from the first node each of the m sets over a separate line; means for receiving at the second node each of the transferred m sets; and means for utilizing the received m sets in accordance with the m start bits.

34. (Currently amended) A hybrid ~~serial/parallel~~ serial to parallel bus interface for use in a synchronous system, the synchronous system having an associated clock, the bus comprising: a data block demultiplexing device having an input configured to receive a data block and demultiplexing the data block into a plurality of nibbles, each nibble having a plurality of bits; an even and odd set of parallel to serial (P/S) converters, each set of P/S converters receiving the nibbles synchronous with a clock rate of the clock, and for converting the nibbles into a serial data; a first set of i multiplexers for transferring the even P/S converters set

serial data on a positive edge of the clock over i lines and the odd P/S converters set serial data on a negative edge of the clock over i lines; a second set of i demultiplexers for receiving the even and odd transferred serial data and sending the even received serial data to an ever buffer and the odd serial data to an odd buffer; the even and odd buffer; an even and odd set of serial to parallel (S/P) converters, the even set of S/P converters for converting the even recited serial data to even parallel data and outputting the even parallel data synchronous with the clock; and the odd set of S/P converters for converting the odd received serial data to odd parallel data and outputting the odd parallel data synchronous with the clock; and a data block reconstruction device for combining the even and odd parallel data as the data block.

35. (Original) The interface of claim 34 wherein each data block has N bits and

36. (Previously presented) The interface of claim 34 wherein the even and odd buffer buffers the even and odd set of S/P converters input so that the even and odd set of S/P converters receive the even and odd received serial data synchronous with the clock.

37. (Original) A method for determining a number of i bus connections required to transfer block data over a bus, each block of the block data having N number of bits, the method comprising: determining a maximum latency allowed for transfer of the block data; determining a minimum number of connections required to transfer the data block with the maximum latency; and determining i with i being a value at least the minimum number of required connections.

38. (Original) The method of claim 37 wherein the i bus connections correspond to i pins on a chip.

39. (Original) The method of claim 38 wherein $1<i<N$.

40. (Currently amended) A system using a bi-directional ~~serial/parallel~~ serial to parallel bus interface comprising: a plurality of lines for transferring data blocks, the plurality of lines numbering less than a number of bits of each data block; a first node sending first data blocks to a second node over the plurality of lines, the first node capable of demultiplexing the data block into a plurality of first nibbles, the plurality of first nibbles numbering a same number as the plurality of lines, each nibble having a plurality of bits; and the second node sending second data blocks to the first node over the plurality of lines, the second

node capable of demultiplexing the data block into a plurality of second nibbles, the plurality of second nibbles numbering a same number as the plurality of lines, each nibble having a plurality of bits.

41. (Original) The system of claim 40 wherein the first node capable of demultiplexing the data block into a plurality of third nibbles, a number j of the third nibbles is less than the number N of lines and transferring the third nibbles over j lines.

42. (Original) The system of claim 41 wherein the second node capable of demultiplexing forth data blocks into K bits, where K is less than or equal to N-j lines, and transferring the fourth data block over K lines.

43. (Original) The system of claim 40 wherein the first node data blocks include gain control information.

44. (Original) The system of claim 43 wherein the second node data blocks include an acknowledgment of receipt of the gain control information.

45. (Original) The system of claim 43 wherein the second node data blocks include information of a status associated with the second node.

46 – 57 (Canceled).